

***What is claimed is:***

1           1. A method of performing natural language generation, the method comprising  
 2 the steps of:  
 3           selecting a reference grammar;  
 4           applying an input dependency tree to a tree choosing module for using a  
 5 stochastic tree model to select syntactic realizations for each node in the derivation tree;  
 6           producing a word lattice for the stochastically selected syntactic realization  
 7 comprising all possible word sequences permitted by the input dependency structure, the  
 8 chosen syntactic realizations, and the reference grammar; and  
 9           choosing a linear precedence output string of least cost from the word lattice.

1           2. The method as defined in claim 1 wherein an extended XTAG grammar is  
 2 selected as the reference grammar.

1           3. The method as defined in claim 1 wherein the Viterbi algorithm is used to  
 2 chose the output string from the word lattice.

1           4. A natural language generator for translating an input dependency syntax tree  
 2 into a natural language output, the generator comprising  
 3           a tree choosing module, responsive to the input dependency syntax tree, for  
 4 stochastically selecting syntactic realizations for each node in the input dependency tree,  
 5 the tree choosing module including a tree model database for use in selection;  
 6           an unraveling module, responsive to the stochastically selected tree-adjointing  
 7 grammar trees created by the tree choosing module and including a predetermined  
 8 reference grammar database for creating from the syntactic realizations a lattice of all  
 9 possible linearizations of said trees using the reference grammar of said database; and  
 10           a linear precedence chooser module for selecting the most likely traversal through  
 11 the lattice as the natural language output of the generator.

1           5. The generator as defined in claim 4 wherein the linear precedence chooser  
 2 module utilizes the Viterbi algorithm to select the most likely traversal path.

1           6. The generator as defined in claim 4 wherein the unraveling module includes a  
2 reference grammar database.

1

1           7. The generator as defined in claim 6 wherein the reference grammar database  
2 comprises an XTAG grammar database.

TOFTO = E37B5260